2 year M. Tech. Course Curricula for Computer Science and Engineering							
M. Tech. I semester (M1)							
S. No.	**Subject Code**	**Subject**	**Core/ Elective**	**L**	**T**	**P**	**Credits**
1	14M11CI112	Advanced Computer Networks	Core	3	0	0	3
2	14M11CI113	Advanced Database Systems	Core	3	0	0	3
3	14M11CI114	High Performance Computer Architecture	Core	3	0	0	3
4	14M11CI115	Advanced Algorithms	Core	3	0	0	3
5		DE-I	Elective	3	0	0	3
6	14M17CI171	Software Systems Lab - I	Core	0	0	4	2
		Total		**15**	**0**	**4**	**17**
List of Electives for DE-I to be updated from time to time			**Core/ Elective**	**L**	**T**	**P**	**Credits**
1	14M14CI132	Artificial Intelligence in Manufacturing	Elective	3	0	0	3
2	14M14CI134	Modern Cryptography	Elective	3	0	0	3
3	14M14MA213	Advance Numerical Techniques	Elective	3	0	0	3
4	14M14PH134	Process Modelling and Simulation of Semiconductor Devices	Elective	3	0	0	3

Course Description

Title of Course: Advanced Computer Networks **Course Code: 14M11CI112**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students must have already registered for the course, "Introduction to Programming" and "Computer Networks".

Objectives: To design and build an advanced secure and low cost Computer Network.

Learning Outcomes: The course will be covered through lecture (power point presentation), assignment sheets (containing questioner from different books will be given on every weekend) and practical labs, some section of the course will include additional reading in the research literature (e.g. research paper on fault-tolerant interconnection network) to allow them to see how the leading edge is advanced in this area of the discipline. Full engagement in the course is a critical part of learning methods appropriate to this area of the discipline. Full engagement in the course is a critical part of learning methods appropriate to this area of study. In practical labs different Network Simulators are used to make all theoretical concepts into practical examples.

Course Contents:

Introduction: Introduction to the Subject and its Relevance. Syllabus. Text Books. Examination. Evaluation and Grading.

Overview and Convergence: Overview of Computer Networks. Convergence of Computer and Telecommunication Networks.

Generic Network Model: Transmission and Switching. Information Types. Bandwidth Considerations. Integration of Voice and Data, Characteristics of Data, Video and Voice Signals.

Switching and Network Types: Packet Switching- Connectionless and Connection Oriented. Circuit Switching and its Comparison with Packet Switching. Characteristics of Data and Telephone Networks.

Layered Architecture: Layering Concepts. 7-Layer Model. Functions of Different Layers. Interfaces and Protocols. Standards.

Transmission Control Protocol and Internet Protocol: Internet Protocol. Addressing Scheme. Protocol Details. IPv4 and IPv6.Internet Control Message Protocol. Transmission Control Protocol and its Features. User Datagram Protocol. Applications.

Data Link and Network Layer: Data Framing. Error Control. High-Level Data Link Control. Network Layer Protocols and Standards.

Local / Metropolitan Area Networks: Local Area Network Types. Topology. Medium Access Control Techniques. IEEE Standards. Physical Level Parameters. Fiber Distributed Data Interface. Distributed Queue Dual Bus. Asynchronous Transfer Mode.

High Speed Networks: High Speed Network Issues. Gigabit Ethernet. Asynchronous Transfer Mode Technology.

Security: Network and Communication Security. Encryption and Decryption Techniques. Firewalls.

Some Advances: Simple Network Management Protocol. Voice Over Internet Protocol. Network Performance. Queuing Theory, etc.

References

1. Andrew S. Tanenbaum, Computer Networks, fourth edition, PHI, 2003.
2. William Stallings, Data and Computer Communication, seventh edition, PHI, 2004.
3. Michael A. Gallo, William M. Hancock, Computer Communication and Networking
4. Technologies, Thomson Press, 2002.
5. Dimitri Bertsekas, Robert Gallager, Data Networks, second edition, PHI, 1992.
6. Douglas E. Comer, David L. Stevens, Internetworking with TCP/IP, Vol I, II and III, PHI, 1998.
7. Oliver Hersent, David Gurle and Jean Pierre Petit, IP Telephony, Pearson, 2000.

Course Description

Title of Course: Advanced Database Systems **Course Code: 14M11CI113**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Student must have knowledge about Database Management Systems.

Objectives: To develop the ability to design, implement and manipulate databases as well as to build Database management systems.

Learning Outcomes:

- Ability to build normalized data bases made a secure application for real time systems.
- Ability to design systems by using Models.
- Ability to develop skills of writing applications by using SQL and PL SQL.
- Ability to understand query optimization techniques.
- Understanding of transaction processing.
- Ability to handle recovery and concurrency issues in the database.
- Ability to design web services will be use in different applications.

Course Contents:

Simple database applications: Single user, Multi User

Multiple types of multiple users: Review of relational databases- EER model, object oriented model, relational algebra, Transforming data models into database design, determination of data requirements for databases. SQL for database processing, advanced SQL, Ranked queries, joins among relations, advanced database manipulation using SQL and PL SQL.

Web enabled database applications: Introduction to PHP, Script elements, Forms, arrays, function and templates, Handling databases, sessions and cookies, File and directory access, Database Access and connectivity

Robust database applications: Transaction state and properties, Atomicity and durability, concurrent transactions, serializability test, Concurrency-Lock based, time stamp based, validation based, and multiple protocols, multiple granularity, deadlock handling.

Robust database applications with large no of concurrent users: Query processing- Measures of query cost, costing of selection operation, sorting, join and other operation, evaluation of expressions, Query optimization- Estimating statistics of expression results, transformation of relational expressions, choices of evaluation plans, materialized view.

Secured Database applications: Database Security - Database security risks, database users and privileges. Authentication, authorization, access control methods, data encryption, Database Recovery- Failure classifications, recovery and atomicity, log based recovery, shadow paging, recovery with concurrent transactions, Buffer management

Multimedia and Mobile databases: Multimedia databases, web-enabled database, temporal databases, Mobile databases.

References

1. Kotler, Philip and Gary Armstrong, Principles of Marketing, 10th Edition, New Delhi, Pearson Education, 2004.
2. Darymple, Douglas J., and Leonard J. Parsons, Marketing Management: Text and Cases, 7th Edition, John Wiley & Sons (Asia) Pte. Ltd., 2002.
3. Kotler, Philip and Kevin Lane Keller, Marketing Management, 12th Edition, New Delhi, Pearson Education, 2006.
4. Winer, Russell S., Marketing Management, 2nd Edition, Prentice Hall, 2003.
5. Dalrymple, Douglas J., and Leonard J. Parsons, 2nd Edition, Wiley Publication, 2000.

Course Description

Title of Course: High Performance Computer Architecture
Course Code: 14M11CI114
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Student must have knowledge about the hardware and software component about the computer and basic of computer architecture and parallel computing.

Objectives: The main aim of this course is intended to develop, implement, and debug assembly language programs that meet stated specifications. To explain bus transactions, memory organization and address decoding, basic I/O interfaces and port addressing. To understand how to control components of a computer system through the use of hardware and software interrupts.

Learning Outcomes: The students shall acquire the generic skills to know about the all aspect of computer like working of memory, memory calculations, design and implement of computer architecture along with analysis of practical aspects.

At the conclusion of the course, following learning objectives are expected to be achieved:

1. You will broaden your knowledge of computer system organization and Architectures.
2. You will gain knowledge in technical aspects of computer system design.
3. You will gain understanding of computer arithmetic both integer and floating point.
4. You will acquire the background for understanding next-generation CPUs.
5. You will learn a computer programming model at a level that enables you to write assembly language programs for the processor meeting given specifications.

Course Contents:

- **Introduction to VHDL:** Implement basic VHDL constructs, Implement modeling structures of VHDL: Behavioral, dataflow, structural, mixed design, use VHDL building blocks: entity architecture, subprograms, package, package declarations, package body, test –Bench, State machine modeling, fault analysis and hazard detection.

- **Functional Organization:** Review the concepts of Computer Architecture – RTL, Micro program, Pipelining and ILP, Memory and I/O. Processor and storage hierarchy, system performance, Performance – Benchmarks, Metrics and their Limitations, Fault tolerance, pipelining timing analysis, and area performance analysis.

- **Performance Enhancements:** Branch Prediction, Dynamic scheduling, Speculative Execution (software/hardware), Superscalar Architecture, Out-of-order Execution, and Multithreading, VLIW and EPIC Architectures, Power Aware Computing, SPEC Mark vs. DTMR cache performance.

- **Multiprocessing:** Amdahl's Law, Multicore and Multithreaded Processors, Flynn's Taxonomy: Multiprocessor Structures and Architectures, Memory Synchronization and Cache Coherence, Interconnection Networks, Programming Multiprocessor Systems, GPU and Special-Purpose Graphics Processors. Case Study – Blue Gene, Road Runner

- **Distributed Systems:** Grids, Cluster Computing, Cloud Computing

References

1. Patterson D A, Hennessy J L, "Computer Architecture – A Quantitative Approach" Elsevier, 3rd Edition.
2. Kai Hwang, "Advanced Computer Architecture – Parallelism, Scalability, Programmability", Mc Graw Higher Education, 1992.
3. Michael J. Flynn, "Computer Architecture-pipelined and parallel processor design", Narosa publication, 8th Reprint, 2008.
4. J. E. Smith, "Characterizing Computer Performance with a Single Number", Communications of ACM, Volume 31, Issue 10 (October 1988), Pages: 1202 – 1206, 1988.
5. D. W. Wall, "Limits of Instruction Level Parallelism", Architectural Support for Programming Languages and Operating Systems, Santa Clara, Pages: 176 – 188, 1991.
6. T. Foster, "Anatomy of the Grid: Enabling Scalable Virtual Organizations", Proceedings of the 7th International Euro-Par Conference Manchester on Parallel Processing, Pages: 1 - 4, 2001.
7. P. Stenström, "A Survey of Cache Coherence Schemes for Multiprocessors", Computer, Volume 23, Issue 6 (June 1990), Pages: 12 – 24, 1990.

Course Description

Title of Course: Advanced Algorithms **Course Code: 14M11CI211**
L-T Scheme: 3-0 **Course Credit: 3**

Prerequisites: Good knowledge of Computer programming, Data Structures and algorithms.

Objectives:

- Strengthen higher level cognitive Skills of analysis, creation and evaluation.
- Strengthen Ability of data abstraction and problem solving using computer
- Strengthen ability to express solution to problem clearly and precisely.
- Strengthen ability to design and evaluate ADTs, nonlinear temporary and persistent data structures and also related algorithms.
- Introduce students to some domain specific data structures and related algorithms in various domains.

Learning Outcomes:

Students will have

1. Deepen the generic skills to design and implement ADTs.
2. Good knowledge to handle non linear Data structures and algorithms for a broad-based set of computing problem in various domains.

Course Contents:

Algorithm Efficiency - Asymptotic Analysis: Growth of Functions. Notions of Theta, Omega, Big-O.

Review of Sorting Algorithms - Popular sorting algorithms – Bubble sort, Insertion sort, Quick sort, Merge sort, Lower bounds for comparison based sorting. Non-comparison based sorting algorithms such as Radix sort, Bucket Sort.

Notion of Correctness in Algorithm design - Need, approaches – systems vs proofs, introduction to faults, modeling, Problem of reaching agreement in presence of faults.

Review of Searching Algorithms - Students will be able to Naive algorithms, Binary search (Importance of preprocessing the data), Hash Tables.

Greedy Algorithms - Activity Selection Problem, Shortest Path Problem, Dijkastra algorithm, Prim's and kruskal Tree.

Dynamic Programming - Matrix Multiplication, Knapsack, Comparrison between Greedy and Dynamic Programming.

String Matching - Naive String Matching, Finite Automata Matcher, Rabin Karp, Knuth Moris Pratt algorithm.

Computational Geometry - Convex Hull, Range Searching, Line segment Intersection Problem, Voronoi diagrams.

Introduction to Amortized Analysis - Motivation, Aggregate Method, Accounting Method, Potential Method.

Introduction to Cryptographic Algorithms - Elementary Number Theory, Fermat's little theorem, RSA public – key cryptosystem.

Primality Testing - Trial Division Method, Fetmat's Theorem, Charmichel Numbers, Miller Rabin test.

NP- Completeness - Notion of P, NP, NP-completeness, reducibility.

Text Book

1. Introduction to Algorithms: Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein ,MIT Press, 2nd Edition, 2001

References

1. Algorithms: S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani, McGraw-Hill Science/Engineering/Math; 1st edition, 2006
2. Data Structures and Algorithms: Alfred V. Aho, J.E. Hopcroft, Jeffrey D. Ullman, Addison-Wesley Series in Computer Science and Information Processing, 1983.
3. Algorithms in C: Sedgewick, 3rd edition. Addison Wesley, 2002.
4. Data Structures and Algorithm Analysis in C: Weiss, Benjamin and Cummings Pub., 1994
5. The Algorithm Design Manual: Steven Skiena, Springer; 2nd edition, 2008.
6. The art of Computer Programming Vol. 1, Fundamental Algorithms: Knuth, Addison-Wesley Professional; 3rd edition, 1997.
7. Fundamentals of Computer Algorithms: Horowitz and Sahni, Computer Science Press, 1978.

Course Description

Title of Course: Software System Lab-I **Course Code: 14M17CI171**
L-T-P scheme: 0-0-4 **Course Credit: 2**

Prerequisite: Students must have good knowledge of C++/JAVA.

Objectives: To develop an understanding of the various components of the UNIX/Linux operating systems from a system programmer's perspective including both the shell and programming interfaces. To develop the ability to use a NS-2 simulator for a designed network.

Learning Outcomes: Students will be able to:

1. Efficient in system programming
2. Creating a process and making a IPC connection
3. Socket programming and System Call functions
4. Comfortable in Network Simulator

Course Contents:

Introduction to UNIX commands and Shell programming, Process model, Process environment, Process creation and termination, Interprocess communication, Process synchronization, Pipes, Named pipes, Introduction to assembly language, Introduction to Compiler, Introduction to NS-2, Mini Project.

Note: There will be three minis Project and each project will be done by different groups.

References

1. Kay A. Robbins, Steven Robbins, "UNIX Systems Programming: Communication, Concurrency, and Threads", Prentice Hall 2003.
2. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, "Compilers: Principles, Techniques, and Tools", Pearson Education 2003.
3. Jeff Duntemann, "Assembly Language Step-by-Step: programming with DOS and Linux" Wiley Dreamtech India Pvt. Ltd., New Delhi, 2003.
4. Arnold Robbins, "UNIX in a Nutshell", O'Reilly 4th edition.
5. David I. Schawartz,"Introduction to UNIX", Prentice Hall, second edition .

2 year M. Tech. Course Curricula for Computer Science and Engineering							
M. Tech. II semester (M2)							
S. No.	**Subject Code**	**Subject**	**Core/ Elective**	**L**	**T**	**P**	**Credits**
1	14M11CI212	Advanced Operating Systems	Core	3	0	0	3
2	14M11CI213	Advanced Software Engineering	Core	3	0	0	3
3	14M11CI214	Multimedia Systems	Core	3	0	0	3
4	14M11CI215	Computer System Performance Analysis	Core	3	0	0	3
5		DE-II	Elective	3	0	0	3
6	14M17CI271	Software Systems Lab - II	Core	0	0	4	2
		Total		15		4	**17**

List of Electives for DE-II to be updated from time to time			**Core/ Elective**	**L**	**T**	**P**	**Credits**
1	14M14CI231	Data Mining & Warehousing Techniques	Elective	3	0	0	3
2	14M14CI233	Enterprise Information Systems	Elective	3	0	0	3
3	14M14CI234	Advanced Computer Graphics	Elective	3	0	0	3
4	14M14CI235	Human Aspects of Software Development	Elective	3	0	0	3
5	14M14CI236	Information System and Security	Elective	3	0	0	3
6	14M14CI131	Analysis and Design of Algorithms	Elective	3	0	0	3

Course Description

Title of Course: Advanced Operating Systems **Course Code: 14M11CI212**
L-T Scheme: 3 **Course Credits: 3**

Prerequisite: Students must have good knowledge of "Introduction to Computer Architecture and Operating System".

Objectives: In this course we will study the advanced components of an operating system, their functions, mechanisms, policies and techniques used in their implementation and examples from popular operating systems like Unix/Linux. The way different modules in the operating system interact and work together to provide the basic services of an operating system.

Learning Outcomes: The students will have a:

1. Detailed knowledge of the concepts of process and shared memory
2. Aware of a variety of approaches to process management and main-memory management
3. Learn the basics behind file systems and input output systems
4. Understand the fundamentals of network and distributed operating systems.
5. Aware of a variety of approaches for Operating system security at process level.

Course Contents:

Overview of OS - Review of Operating Systems principles like Linux, VxWorks, and Comparative analysis.

Design Issues - Layering, Structure: Monolithic, Micro kernel, etc. Merits and demerits of Minix, µCOS etc.

Communication and Synchronization - Synchronization mechanisms. Process deadlocks. Resource models. Clock synchronization, Local and Global states. Inherent limitations of distributed operating systems. Event ordering. Timestamps.

Concurrency control algorithms - Distributed mutual exclusion. Token and non-token based algorithms. Comparative performance analysis. Concurrency control.

Device Driver - Char, Block, Network, Serial and USB device drivers. Case studies of Keyboard device driver, HDD device driver, and LAN card drivers. New approach for Hot pluggable USB device drivers for all kind of devices.

File Systems - File handling and recovery mechanisms. Case study of FAT, EXT2FS etc.

Security - Authentication mechanisms, Nature of Attacks, Secure network communications, Kerberos, etc.

Design Studies - Linux, VxWorks, MicroC/OS-II, Spin, Aegis, Mach, etc.

Text Books

1. Advanced Concepts in Operating Systems: M. Singhal, N. G. Shivaratri, 1st Ed., Tata McGraw-Hill, 1994.

References

1. Linux Device Drivers : Alessandro Rubini, 3rd Ed., O'Reilly, 2005.
2. Real-Time Systems: C. M. Krishna, K. G. Shin, 1st Ed., McGraw-Hill, 1997.
3. MicroC/OS-II, The Real-Time Kernel: J. J. Labrosse, 2nd Ed., CMP Books, 2002.
4. Real-Time Systems: J. W. S. Liu, 1st Ed., Addison-Wesley, 2000.
5. The Linux Kernel Module Programming Guide: Peter Jay Salzman, Michael Burian, Ori Pomerantz, 2.6.1 Edition, Linux Documentation Project, 2005.
6. Distributed Operating Systems: Tanenbaum, A. S 1st Ed., Prentice-Hall, Englewood Cliffs, NJ, 1995.

Course Description

Title of Course: Advance Software Engineering **Course Code: 14M11CI213**
L-T Scheme: 3 **Course Credits: 3**

Prerequisite: Students must have good knowledge of "Software Engineering and Project Management".

Objectives: To engineer good quality software from its specification.

Learning Outcomes:

- Familiar with processes of Advanced Software Engineering
- Awareness about handling the complexities that may arise in various stages of SDLC
- Generating test cases for software testing
- Computer Aided Software Engineering
- Aspect of Quality in Software Development

Course Contents:

Review of software engineering: Requirement Engineering, Software Process Models, Analysis and Design, UML Diagrams.

Software Reengineering: Software reengineering, forward engineering, reverse engineering, program comprehension.

PSP and TSP: Time management, Tracking time, Product planning, Project Plans, Defect finding, projecting a defect, Quality: Product and Process quality.

Software Metrics: Introduction to Metrics, challenges in Metrics, classification of Metrics. Product Metrics: Metrics for Analysis, Metrics for design, Metrics for code, Metrics for Testing. Process Metrics: Size, cost, effort Estimation, COCOMO Model. Project Metrics: Metric for Quality.

Agile Methods: Agile development, Classification of methods, Agile principals, Agile project management, SCRUM, extreme Programming.

Advance software Testing: Object Oriented Testing , GUI Testing , Real Time system Testing, Automated Testing, cyclomatic complexity, black box testing, data flow testing, graph based testing, regression testing.

Software Security: Software Threats, Software Piracy and authentication, architecture and design of secure software, process of building secure software.

Text Book

1. Software Engineering: A practitioner's approach: Roger S. Pressman, McGraw-Hill Publications (Sixth Edition)

References

1. Software Testing Techniques, B. Beizer (Second Edition)
2. Agile and Iterative Development: A Manager's Guide, Craig Larman
3. Software Engineering, Sommerville, Addison Wesley
4. Design Patterns, Erich Gamma, Johnson, Pearson Education
5. IEEE International Conference on Software Maintenance (ICSM)
6. ACM Transactions on Software Engineering Methodology.
7. IEEE Transactions on Software Engineering.

Course Description

Title of Course: Multimedia System **Course Code: 14M11CI214**
L-T Scheme: 3-0 **Course Credits: 3**

Objectives: Understand concepts, algorithms and design principles underlying modern multimedia systems.

Learning Outcomes: Students will have

1. Deepen knowledge of audio and video stream.
2. Good knowledge of different encoding and compression scheme like MPEG, MPEG4.
3. Ability to deal with GPRS, 3G/4G wireless systems.

Course Contents:

Multimedia Programming - Term Multimedia, Term Media, Characterizing Data Streams. Advanced Object Oriented Programming, Reusability, Synchronization, Expandability, Maintainability.

Digital Audio and Video Technology - Audio Technology : Audio Representation on Computers, Speech Output, Speech Input, Speech Transmission. Audio Compression: Differential pulse code modulation, Adaptive differential PCM, Adaptive predictive coding, Linear Predictive Coding, Perceptual Coding, Audio Coders. Text Compression: Static Huffman coding, Dynamic Huffman coding, Arithmetic coding, Lempel-Ziv coding. Image Compression : Graphics Interchange format, Tagged image file format, JPEG Video Compression: MPEG, MPEG4, Audio and Video Synchronization

Multimedia Database Management Systems - Multimedia object storage, File Retrieval structures, Disk Scheduling , Metadata for Multimedia, Multimedia Data Access, Multimedia Information Modeling, Object - Oriented Modeling, Querying Multimedia Databases.

Multimedia Communication - Standards for multimedia communications. TCP – based system for multimedia streaming, Peer – to – Peer streaming topologies, Distributed Multimedia applications, live / on-demand broadcasting, server bandwidth, distribution control and privacy protection for Internet media delivery, High-End Interactive Television Terminals. Wireless IP networks, wireless video communication, 3G/4G wireless systems. Multimedia applications over Multihop wireless links, World Wide Web.

Multimedia Content Analysis - Analysis of individual images, text recognition, similarity based searches in image databases, Audio Analysis, Video Analysis.

Text Book

1. Multimedia Applications: Ralf Steinmetz and Klara Nahrstedt, Springer, 2007.

References

1. Media Coding and Content Processing: Ralf Steinmetz and Klara Nahrstedt, PHI, 2005.
2. Multimedia Database Management Systems, B. Prabhakaran, Springer, 2007.
3. Multimedia Communications Applications, Networks, Protocols and Standards: Fred Halsall, Pearson Education, 2008.
4. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP).

Course Description

Title of Course: Software System Lab-II **Course Code: 14M17CI271**
L-T-P scheme: 0-0-4 **Course Credit: 2**

Prerequisite: Students must have good knowledge of Java.

Objectives: The course will help you in learning J2EE technology along with Network simulators. It aims for the students to gain experience in Advance Data Structure, Advance Algorithm and ADBMS programming with writing efficient, portable, correct, and large scale programs in the Windows environment using j2ee technology. To give the student a solid background of design and performance issues associated with ADS, AA, and ADBMS using J2EE.

This course will also help you in the learning and understanding of the layered network architecture with its protocols. The syllabus is completely practical oriented. The emphasis will be to introduce you to the latest in the field of networks and communications. The lab exercises will consist of experiments to be performed on simulators:

- NS2
- Omnetpp 4.0

Learning Outcomes: Student will be able to learn and understand

1. Complete J2EE Technology.
2. Advanced Data Structure, Advance Algorithm Programming
3. Developing enterprise application using J2EE
4. Designing and Implementing of network models

Course Contents:

Java Servlet Programming, JSP Programming: JSP and Servlet Session Handling, JSP Directives, JSP Standard Actions, JSP Error Handling, JSP with JDBC Access, JDBC API, Maintaining Users Session log files

Enterprise Java Bean

Entity Bean, Session Bean, Message Driven Beans, Web Service, Development

Invoking Java Cryptography API

Symmetric –Asymmetric Encryption API, Digital Signature, Message Integrity API

Introduction: Familiarization with the ns2 and Omnetpp 4.0 simulation environment, basics of queuing

Node Creation and Data transfer: Two node, multi node, simplex communication, duplex communication, calculation of network metrics

Designing a Network and Data transfer: Data loss, error correction, routing on different network topologies.

Applying Transport Protocols: Test TCP and UDP impact on network bandwidth; testing on different TCP flavors; Introduction to wireless transport protocols

Building Applications on top of Network model: Implementation of different application layer protocols mainly FTP, HTTP, DNS etc.

Text Books

1. Core Servlets and Java Server Pages: Volume 1: Core Technologies Java 2 Platform Enterprise Edition Series, Marty Hall and Larry Brown.

Reference Books

1. Enterprise Java Beans: O'Relly, Head First.
2. Complete Reference of J2EE: Tata McGraw-Hill, Herbert Schlitz.
3. The complete Reference of JAVA, Fourth Edition, Herbert Schlitz

Web References

1. http://www.isi.edu/nsnam/ns/
2. http://www.omnetpp.org/

Course Description

Title of Course: Data Mining and Data Warehousing **Course Code: 14M14CI231**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students are expected to have a grasp of the fundamentals of computer system, including a basic understanding of the operation of the computer, especially Database Management Systems. In addition, students are expected to know application development environment and programming concepts.

Objectives: The main objective of this course is to provide students with the basic data warehousing, data and Web mining concepts and applications that can enable them to set up and manage an industrial data warehousing and data mining system.

Learning Outcomes: Students will develop the following-

1. Ability to do Conceptual, Logical, and Physical design of Data Warehouse
2. Have a conceptual and practical understanding of the data and web mining.

Course Contents:

Data warehousing, Data mining, Web Crawling and Web search Introduction.

Requirement analysis, Dimensional modeling, Design, Project management, Building and Testing.

User data access management, Online analytical Process models, Query Access Architectures, OLAP, MOLAP, ROLAP.

Data preprocessing, Data extraction, transformation and loading.

Meta data and Metadata Environment.

Data warehouse Quality metrics and Security.

Linear regression, multiple regression, Usability and Complexity Analysis of Bayesian algorithm, Nearest Neighbor algorithm, Decision Tree based algorithm, Rule based Algorithm.

Usability and Complexity Analysis of Agglomerative Hierarchical Algorithm, k-means Partitioning Algorithm, self organizing maps, Multidimensional scaling.

Usability and Complexity Analysis of Apriori Algorithm, Sampling Algorithm, Partitioning, Using multiple minimum supports , Rough set approach

Text Books

1. Data Mining, Concepts and Techniques: Jiawei Han and Micheline Kamber, Elsevier 2nd Edition.

References

1. Data Mining: Introductory and Advanced Topics: Margaret H. Dunham, Prentice Hall.

2. Data Warehousing, Data Mining and OLAP: Alex, Berson, Stephen J. Smith, Tata McGraw-
3. Hill, 2004.
4. Mining the Web : Discovering knowledge from hypertext data: Soumen Chakrabarty, Morgan
5. Kaufmann
6. Building the Data Warehouse: Inmon W.H., Wiley
7. Data Warehousing in the Real World: Anahory S. and Murray D, Addison-Wesley
8. The Data Warehouse Toolkit: Kimball R. and Ross M ,Wiley
9. Web Warehousing and Knowledge Management: Mattison R., Tata McGraw-Hill.
10. Principles of Data Mining: David Hand, Heikki Mannila and Padhraic Smyth;PHI
11. Data mining and statistical analysis using SQL: Pujari, Arun K., Universities press
12. Introduction to Data Mining: Pang-Ning Tan,Michael Steinbach,Vipin Kumar,Pearson
13. Web Mining: from web to semantic web: Beltina Berendt, Dunja Mladnic, Springer,2003

Journals

1. Transactions on Database Systems (ACM)
2. IEEE Transactions on Knowledge & Data Engineering
3. The VLDB Journal The International Journal on Very Large Data Bases
4. IEEE International Conference on Data Mining

Course Description

Title of Course: Computer Systems Performance and Analysis
Course Code: 14M14CI232
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students must have knowledge of "High Performance Computer Architecture"

Objectives: Impart a comprehensive idea to design, procure or use computer systems by knowing the analytical simplicity of how to evaluate them.

Learning Outcomes: It is expected that at the end of the course a student must be able to comprehensively analyze the performance and even evaluate them for computer systems. To achieve this goal one will gain sufficient statistical skills for performance analysis which is a key step in procurement of new computer systems including processors, languages, and operating systems, networking architectures or database systems. The student will even be able to simulate and model out a system under performance evaluation.

Course Contents:

An overview of performance evaluation: Introduction, Common mistakes and how to avoid them, Selection of techniques and metrics.

Measurement techniques and tools: Types of workloads, Art of work load selection, Workload characterization techniques, Monitors, Program execution monitors and accounting logs, Capacity planning and bench marking, The art of data presentation, Ratio games.

Probability theory and statistics: Summarizing measured data, Comparing system using sample data, Simple linear regression models, other regression models.

Experimental design and analysis: Introduction to experimental design, 2^k Factorial design, 2^kr Factorial designs with replication, 2^{k-p} Fractional factorial design, One factor experiments, Two-Factor full factorial design without replications, Two-Factor full factorial design with replications, General Factor full factorial design with k factors.

Simulation: Introduction to simulation, Analysis of simulation results, Random-number generation, testing random-number generators, Random–variate generation, commonly used distributions.

Queuing Models: Introduction to queuing theory, Analysis of a single queue, queuing networks, operational laws, Mean value analysis and related techniques, Convolution Algorithm, Hierarchical decomposition of large queuing networks.

Text Books

1. **"**The Art of Computer Systems Performance Analysis", Techniques for Experimental Design, Measurement Simulation and Modeling by Raj Jain, Wiley India Pvt Ltd.

References

1. "Measurement and Modeling methods For Computer System Performance" Barnes, M.F., Input Two-Nine, Surry, UK.

Course Description

Title of Course: Enterprise Information System **Course Code: 14M14CI233**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Introduction to basic Java Programming, Data Base Management System and Software Engineering.

Objectives: The main objective of this course is to provide students with the basic enterprise oriented data processing with Java programming.

Learning Outcomes: Students will develop the following:

- Ability to do Conceptual, Logical, and Physical design of any Enterprise Information System
- Have a conceptual and practical understanding of the Mobile Based Enterprise Application.

Course Contents:

Introduction to Integrated Enterprise Information System: Enterprise Architecture Models, Frameworks, Classification of Information Systems, Functional Area Application-Finance, Marketing, HR, Software Enterprise, VLSI Enterprise.

Introduction to JAVA: Collection Interfaces, The Collection Classes, The Legacy Classes and Interfaces, Programming Advanced Data Structure using Collection APIs.

Enterprise Application with J2EE: J2EE Architecture, Servlet Lifecycle, Servlet Classes, Threading models, Elements of JSP, Database Access with JDBC, JSP & XML, JAVA Web Services, JAVA Message Services.

Developing Enterprise Application using Enterprise JAVA Beans: Three Tier Architecture, Message Driven, Entity Bean, and Session Bean.

Social, Legal, Ethical & Security issues in EIS: Security Architecture, Security Risks and Threats Security Policies, JAVA Cryptography.

Mobile Based Enterprise Application using Java2 Micro Edition: Configuration, Profiles, Architecture, Development Kits, Creating, compiling, running and deploying MIDlet suit, J2ME Wireless Toolkit.

Enterprise Information System for software Enterprise: Requirement Management system, Online Defect Tracking System for Software Engineers, Software Testing enterprise System, Documentation Version Management System, Change Management System, Software Project Management System.

Text Books

1. The Complete Reference: JAVA, Herbert Schildt, Tata McGraw- Hill
2. The Complete Reference: JSP 2.0, Phil Hanna, Tata McGraw-Hill

References

1. Database system concepts: Henry F Korth, Abraham Silberschatz, S. Sudurshan, 4th Edition, McGraw-Hill.
2. An Introduction to Database Systems: C. J. Date, Pearson Education.
3. Data Management: databases and organization: Richard T. Watson, Wiley.
4. Data Modeling Essentials: Graeme C. Simxion, Dreamtech.
5. Oracle 9i manuals.
6. ACM Transactions on Database Systems.
7. International Journal of Enterprise Information Systems, Published by Idea Group Inc. ISSN (printed): 1548-1115. ISSN (electronic): 1548-1123.
8. Head First EJB: Kathy Sierra, Bert Bates, O'Reilly.
9. Head First-JSP and Servlets: Basham, Bates, Sierra, O'Reilly, 2nd Edition.
10. Advances in enterprise information systems, Information Systems Frontiers, Springer Netherlands, 1387-3326 (Print) 1572-9419 (Online), Vol. 10, No.5, Nov 2008.

Course Description

Title of Course: Advanced Computer Graphics **Course Code: 14M14CI234**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students must have-

- Knowledge of computer graphics fundamentals (CS426 or equivalent)
- Ability to program in C/C++
- Enthusiasm!

Objectives:

This course will study advanced topics in computer graphics. The focus will be on learning recent methods in rendering, modeling, and animation. The course is appropriate for students who have taken CS426 (or the equivalent) and would like further exposure to computer graphics.

Course Contents:

Introduction: Introduction, Direct Illumination, rendering concepts, lighting, reflectance

Monte Carlo: Global Illumination, rendering equation, comparison of solution methods**,** Monte Carlo Path Tracing I: Monte Carlo integration, path tracing

Radiosity: Monte Carlo Path Tracing II: sampling strategies, irradiance caching, form factors, solution methods, meshing

Visibility & Texture: visibility events, discontinuity meshing

Texture: texture mapping, resampling

Image-based modeling rendering: Managing Scene Complexity: occlusion culling, detail elision, imposters, plenoptic function, image-based representations

3D Modeling: Object Representations: overview of 3D model representations, Mesh Representations: mesh data structures, simplification

Triangle Meshes: multi-resolution meshes, progressive meshes, view-dependent simplification, Mesh Processing: compression, streaming

Surface Modeling: Parametric surfaces: splines, piecewise polynomial surfaces, subdivision surfaces: subdivision schemes.

Volumetric Modeling: Implicit surfaces: blobby models, skeletons, variation implicit surfaces, Volumetric Representations: voxels, volume graphics

Kinematics: articulated figures, inverse kinematics, space-time constraints, Motion Capture: processing motion capture data, retargeting motion

Dynamics: Passive Dynamics: particle systems, spring-mass systems, Active Dynamics: controllers, learning, planning,

Wrapup: review and discussion

Text Book

1. Alan Watt and Mark Watt, Advanced Animation and Rendering Techniques: Theory and Practice, Addison-Wesley, NY, 1992, ISBN 0-201-54412-1.

Course Description

Title of Course: Human Aspects of Software Development **Course Code: 14M14CI235**
L-T Scheme: 3-0 **Course Credits: 3**

Objectives: This course will investigate the research on the human aspects of software development. The focus will primarily be on individual software development, such as what is known about people programming, debugging, testing, and understanding code. We will cover studies of programmers, and tools that have been shown to be effective for programmers.

Course Contents:

HCI techniques relevant to studying software development (contextual inquiries, field studies, surveys, lab studies, etc.)
Studies of novice programmers (barriers, performance, etc.)
Studies of professional programmers (programmer variability, cultural differences, actual practices, questions asked)
Studies of the usability of APIs
Studies of processes for programming (Agile, Peer, collaboration, test-driven development, etc.)
Tools for novice programmers
Tools for making it easier to understand code (Visualization, analyses, etc.)
Tools for making it easier to write programs (IDEs, etc.)
Programming Languages and Environments that have good usability properties
Tools for making it easier to debug programs
Tools focused on "end-user" development of code (for people who are not professional programmers)
End-User Software Engineering (EUSE)
Tools for making it easier to capture and use Design Rationale
Documentation tools for making it easier to understand APIs

References

2. John M. Carroll, “Human Computer Interaction in the New Millennium”, Pearson Education.
3. Saara Baase, “Wings of Fire: Social, Legal, and Ethical Issues in Computing”, 2nd edition, Prentice Hall, 2003
4. A.N. Tripathi, Human Values, revised second edition, New Age, 2006
5. ACM Computer and society magazine, 1972 to 2005.
6. IEEE Technology & Society Magazine, 1988 onward.
7. Software Engineering Code of Ethics and Professional Practice (Version 5.2), ACM/IEEE-CS.

8. Proceedings of the ACM SIGCPR conference on Computer personnel research.
9. Proceedings of the ACM SIGMIS-CPR conference on Computer personnel research.
10. Proceedings of the conference on Ethics in the computer age, ACM, 1994.
11. Journal of Ethics and Information Technology, Springer.
12. Journal of Business Ethics, Springer.
13. Journal of Ethical Theory and moral practice, Springer.
14. Journal of Value inquiry, Springer.
15. Journal of Human studies, Springer.
16. The Journal of ethics, Springer.
17. Journal of Empirical Software engineering, Kluwer Press.

Course Description

Title of Course: Information System and Security **Course Code: 14MI4CI236**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Since the tools used in class are compiled and run on a Linux system, Linux or UNIX system experience is helpful, but not necessary. A solid background in networking concepts will greatly aid in comprehension. This is an intense class that covers many topics.

Objectives: Main objective of this course will be to provide a basis for security planning when while designing new applications or expansion of current network. It describes user responsibilities, such as protecting confidential information and creating nontrivial passwords. Security policy should also be described to monitor the effectiveness of security measures. Such monitoring helps the user to determine whether someone may be attempting to circumvent the safeguards.

Learning Outcomes:

- Upon successful completion of the courses in this discipline, the student will have acquired the following knowledge and skills:
- Demonstrate an appreciation of the IT career field and the need to be lifelong learners. Increase the ability to identify new learning requirements and to learn independently.
- Demonstrate oral and written communication skills and increase ability to be effective team members.
- Demonstrate attitudes that are beneficial to maintaining the security of a computer/network system and assisting people to use that system or network.
- Demonstrate confidence to work independently to setup and maintain computer and networking systems.
- Demonstrate techniques to anticipate and prepare for a variety of unknown situations that might impact the operation of a computer system or network.
- Demonstrate understanding of how computers communicate with each other and the methods employed to assure that the communication is reliable.
- Participate in a structured internship based in the workplace and receive real world, hands on experience.

Course Contents:

Background: Review of networking technologies, Network security threats and counter measures

Cryptography and trust models: Secret key cryptography, Hashes and message digests, Public key cryptography

Network and system security applications: Authentication and security handshakes pitfalls, IP security, Web and E-commerce, Attacks to routing infrastructures and counter measures

Systems and Application Security: OS Security Models, Principals of Web and regular App security, database security

Network Surveillance: Network Attack Traceback and Attribution
Intrusion Detection and Response: What and Why's of IDS
Non repudiation: Anonymity/Pseudonymity/P3P
Case Studies

References

1. Network security: Private communication in a public world by Kaufman, Perlman, and Speciner
2. Network security essentials: applications and standards by William Stallings.
3. ACM Transactions on Information and system security
4. IEEE Press Computer Security and Privacy
5. International Journal of applied cryptography
6. International Journal of Electronic Security and Digital Forensics
7. International Journal of Information and Computer Security
8. International Journal of Security and Networks

Course Description

Title of Course: Advance Numerical Techniques **Course Code: 14M14MA213**
L-T Scheme: 3-0 **Course Credits: 3**

Objectives: This course has been prepared for students pursuing M.Tech and Ph.D programme. The main role of Numerical Methods is to approximate the accurate solutions of complex problems of engineering as well as any mathematical science. This course offers complete procedures for solving numerically various kinds of problems that appear in engineering and science in galore.

Learning Outcomes: At the end of this course students will be acquainted with the basic concepts, techniques and mathematical rigor in numerical methods fully and their applications in a variety of contexts reflecting on real life problems.

Course Contents:
Newton-Raphson Method for two or more variables, Muller's Method, Graeffe's Root Squaring Method, Lin-Bairstow's Method.

Least Square Method, Iterative methods for A x= b, SOR method, Jacobi Method, Householder's Method for eigenvalues and vectors.

Interpolation using Divided Differences, Hermite Interpolation, cubic splines. Orthogonal polynomials, GraSchmidt orthogonalization process, Chebyshev polynomials, economization of power series, numerical integration, Romberg integration, Newton-Cotes integration formulae, Gaussian Quadrature.

Single step- methods, multi step methods, predictor-corrector method for initial value problems. Finite difference methods for elliptic (Laplace equation), parabolic (heat equation) and hyperbolic equations (wave equation).

Method of weighted residuals, Rayleigh-Ritz method, finite element method for one dimensional problems.

References

1. Smith, G.D., Numerical Solution of Partial Differential Equations: Finite difference methods, Third Edition, Oxford University Press 1985.
2. Gerald, C.F., Applied Numerical Analysis, 6th Edition, Addition Wesley, 2002.
3. Conte, S.D and deBoor, C., Elementary Numerical Analysis, An algorithmic Approach,3rd edition, McGraw-Hill, New York,1980.
4. Jain, M.K., Iyenger, S.R.K.and Jain R.K., Numerical Methods for Scientific and Engineering Computation, New Age, 2000.

42 year M. Tech. Course Curricula for Computer Science and Engineering							
M. Tech. III semester (M3)							
S. No.	**Subject Code**	**Subject**	**Core/ Elective**	**L**	**T**	**P**	**Credits**
1		DE-III	Elective	3	0	0	3
2		DE-IV	Elective	3	0	0	3
3	14M19CI392	Seminar-I	Core	0	0	4	2
4	14M19CI391	Dissertation Part I*	Core	--	--	--	12
		* to be continued in Semester IV					
		Total					**20**
List of Electives for DE-III & DE-IV to be updated from time to time			**Core/ Elective**	**L**	**T**	**P**	**Credits**
DE-III							
1	14M14CI331	Advanced Network Management	Elective	3	0	0	3
2	14M14CI332	Image Processing & Applications	Elective	3	0	0	3
3	14M14CI333	Real Time Operating System	Elective	3	0	0	3
4	12M1GCI335	Cognitive Sciences	Elective	3	0	0	3
5	14M14CI336	Queueing Networks	Elective	3	0	0	3
DE-IV							
6	14M14CI334	Computation Theory and Applications	Elective	3	0	0	3
7	14M14CI335	Computational Intelligence	Elective	3	0	0	3
8	14M14MA231	Mathematical Modelling & Simulations	Elective	3	0	0	3
9	14M14PH131	Quantum Computation & Quantum Cryptography	Elective	3	0	0	3

Course Description

Title of Course: Advanced Network Management **Course Code: 14M14CI331**
L-T Scheme: 3-0 **Course Credits: 3**

Objectives: The course covers advanced aspects of managing enterprises computer networks in a modern networked environment. It also dwells into details of Remote Monitoring, Simple Network Management Protocol, Network Management of MPLS networks.

Learning Outcomes: After completing this course, students will be able to use network management tools, systems and applications in an organization. They should be able to explain the use of network management protocol, remote monitoring, and MPLS network management.

Course Contents:
Overview of Network Management, Managing Enterprises Networks, Network Implementation and Management, Configuration: Infrastructure Components, SNMP, Network Management Problems and Their Solutions, RMON1, RMON2, Management of MPLS Networks, Desktop Management.

Text Books

1. J Richard Burke, Network Management Concepts and Practice: A Hands-on Approach, PHI.
2. Stephen B Morris, Network Management, MIBs and MPLS Principles, Design and Implementation, Pearson Education.

Course Description

Title of Course: Image Processing and Applications **Course Code: 14M14CI332**
L-T Scheme: 3-0 **Course Credits: 3**

Objectives: This course aims to provide every student with a working knowledge of sophisticated methods and techniques for collecting, processing and analyzing remotely sensed data; as well as the theory and practice of undertaking remote sensing projects. Throughout the course, emphasis will be placed on image processing, image analysis, image classification, remote sensing and GIS data integration, and applications of remote sensing in geographical analysis and environmental monitoring.

Course Contents:

Introduction and Digital Image Fundamentals: Digital Image Representation, Fundamental Steps in Image Processing, Elements of Digital image processing systems, Sampling and quantization, some basic relationships like neighbors, connectivity, Distance measure between pixels, Imaging Geometry.

Image Transforms: Discrete Fourier Transform, Some properties of the two-dimensional Fourier transform, Fast Fourier transform, Inverse FFT.

Image Enhancement: Spatial domain methods, Frequency domain methods, Enhancement by point processing, Spatial filtering, Lowpass filtering, Highpass filtering, Homomorphic filtering, Colour Image Processing.

Image Restoration: Degradation model, Diagnolization of Circulant and Block-Circulant Matrices, Algebraic Approach to Restoration, Inverse filtering, Wiener filter, Constrained Least Square Restoration, Interactive Restoration, Restoration in Spatial Domain.

Image Compression: Coding, Interpixel and Psychovisual Redundancy, Image Compression models, Error free comparison, Lossy compression, Image compression standards.

Image Segmentation: Detection of Discontinuities, Edge linking and boundary detection, Thresholding, Region Oriented Segmentation, Motion based segmentation.

Representation and Description: Representations schemes like chain coding, Polygonal Approximation, Signatures, Boundary Segments, Skeleton of region, Boundary description, Regional descriptors, Morphology.

Recognition and Interpretation: Elements of Image Analysis, Pattern and Pattern Classes, Decision-Theoretic Methods, Structural Methods, Interpretation.

Text Books

1. Rafael C. Gonzalez & Richard E. Woods, “Digital Image Processing.

2. Rafael C. Gonzalez & Richard E. Woods, “Digital Image Processing using MATLAB.
3. Sonka, Hlavac & Boyle, “ Image Processing, Analysis and Machine Vision”, Thomson Engineering

References

1. Rosefield Kak, “Digital Picture Processing”,
2. W.K. Pratt, “Digital Image Processing”,
3. A.K. Jain, “Fundamental of Digital Image Processing”, PHI

Course Description

Title of Course: Real Time Operating System **Course Code: 14MI4CI333**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students must have already registered for the course "Operating System"

Objectives:

The course focuses on different issues associated with using a single processor computer system to implement a real-time system as well as the control and data dependency among tasks in multiprocessor systems. Specifically the student would study the operating system issues, scheduling, communication protocols and synchronization of execution of tasks on different processors.

Learning Outcomes: At the end of the course, the student shall be able to:

1. Define the concepts of real-time systems.
2. Recognize the characteristics of a real-time system.
3. Identify and determine important software engineering principles for real-time system development.
4. Produce an architectural design of a real-time system

Course Contents:

Basic Concepts: Hard versus soft Real Time System, A reference model of Real Time System, Real Time applications.

Real Time Scheduling: Clock driven scheduling, Priority driven scheduling of periodic tasks, Scheduling, A periodic and sporadic jobs in priority driven systems

Real Time Operating Systems: Time services and scheduling mechanisms, Basic Operating System Functions, Processor reserves and Resource Kernel, Open System Architecture, Overview of several commercial and general purpose operating systems.

Multiprocessor and Distributed real time systems: Model of multiprocessor and distributed systems, Multiprocessor scheduling, Resource access control and synchronization.

Resources Access Control: Assumptions on resource and their uses, Resource Contention and Resource Access Control, Non preemptive critical sections, Basic priority inheritance protocol, Basic priority ceiling protocol, stack based, Priority ceiling protocol.

Real time Communications: Model of Real Time Communications, Priority based service disciplines for switched networks, weighted Round Robin service disciplines, Medium Access Control Protocols of Broadcast Networks, Internet and Resource Reservation Protocols, Real Time Protocol, Communication in Multicomputer Systems.

Text Books

1. Jane W.S. Liu, "Real Time Systems", Pearson Education

References

1. C. M. Krishna and Kang G. Shin, "Real Time System", McGraw Hill.
2. Yann Hang Lee and C. M. Krishnan, "Readings in Real Time Systems" IEEE Computer Society Press.

Course Description

Title of Course: Computation Theory and Applications **Course Code: 14MI4CI334**
L-T Scheme: 3-0 **Course Credits: 3**

Objectives: Complexity theory considers not only whether a problem can be solved at all on a computer, but also how efficiently the problem can be solved. Two major aspects are considered: time complexity and space complexity, which are respectively how many steps does it take to perform a computation, and how much memory is required to perform that computation.

Course Contents:

Review of Theory of Computation: Regular Expression, Context free grammar, Pushdown automata, Turing Machines

Finite state Machine Patterns: State object pattern, Basic finite state machine pattern, State driven transition, Owner driven Transition, Layered structure, Interface Organization, Mealy ,Moore, Hybrid FSM, Exposed FSM, Encapsulated FSM, static State Instantiation FSM, Dynamic state Instantiation FSM Pattern, Timed Automata

Formal software specification methods: Introduction, Specification and requirements, Specification languages and Applications

Tools: Language Emulator, JFLAP, Turing Machine simulator. RAISE

References

1. Hoproft J.E. and Ullman J.D., Introduction to Automata Theory, Languages and Computation, 2nd Edition, Pearson Education.
2. Moret B., the Theory of Computation, Pearson Education, 2001.
3. Mishra, K.L.P. And Chadrasekaran N., the Theory of Computer Science, 2nd Edition, Prentice Hall Of India, 2001.
4. Martin John C., Introduction to Languages and the Theory of Computation, 3rd edition, McGraw-Hill Inc., New York, 2003.
5. Linz, P, an Introduction to Formal Languages and Automata, Narosa Publishing House, 2003.
6. Sipser, M., Introduction to the Theory of Computation, PWS Publication.
7. Savage J.E., Models of Computation, Addison-Wesley, 1998.
8. ACM Transactions On Computation Theory, ACM Journal on Theory of Computation, ACM SIGACT
9. Alfred V. Aho, Jeffrey D. Ullman, Compiler Design, Pearson Education.
10. Alfred V. Aho, Jeffrey D. Ullman, Monica S. Lam, Ravi Sethi, Compilers (Principles, Techniques and Tools), Person Education.
11. Michael Hutch and Mark Ryan, Modeling & Reasoning about systems(Logic in Computer Science), Cambridge.
12. Hung Dang Van, Chris George and Richard Moore, Specification Case studies in RAISE,UNU-IIST.

Course Description

Title of Course: Computational Intelligence **Course Code: 14M14CI335**
L-T: 3-0 **Course Credits: 3**

Objective: Introducing concepts, models, algorithms, and tools for development of intelligent systems. Example topics include artificial neural networks, genetic algorithms, fuzzy systems, swarm intelligence, ant colony optimization, artificial life, and hybridizations of the above techniques. Students will be able to sense these techniques from a machine learning perspective. This domain is called Computational Intelligence, and is a numerical interpretation of biological intelligence.

Learning Outcome: On the completion of this course, the student will have:

- An understanding of fundamental computational intelligence and machine learning models.
- Implemented neural networks, genetic algorithms, and other computational intelligence and machine learning algorithms.
- Applied computational intelligence and machine learning techniques to classification, prediction, pattern recognition, and optimization problems.

Course Contents:

Computational intelligences, agents, example application domains, Representation and reasoning systems, Datalog, syntax and semantics, variables, queries, answers, recursion. Proofs, soundness, completeness, top-down and bottom-up reasoning, function symbols, Searching, graphics, generic search engine, blind search strategies, heuristic search, A* search. Pruning the search space, search direction, iterative deepening, dynamic programming, constraint satisfaction, consistency algorithms, hill climbing, randomized algorithms. Knowledge representation issues, defining a solution, choosing a representation, semantic networks, frames, primitive and derived relations. Equality, inequality, unique names assumption, complete knowledge assumption, negation as failure. Actions and planning. STRIPS representation, situation calculus, forward planning, resolution and planning. The STRIPS planner, Midterm, Regression Planning. A building situated robots Robot Architectures

Textbooks:

1. Computational Intelligence : Concepts to Implementations by Eberhart & Shi

Reference Books:

1. Introduction to Genetic Algorithms by Melanie Mitchell
2. Handbook of Genetic Algorithms by Davis
3. Machine Learning by Tom Mitchell

Course Description

Title of Course: Queueing Networks **Course Code: 14M14CI336**
L-T Scheme: 3-0 **Course Credits: 3**

Objectives: This course is designed for providing support of mathematical modelling to the concepts of computer networks.

Learning Outcomes: Students will be able to analyze a number of important modeling techniques in the form of "Algorithms". The students practicing as computer system analysts are relatively skilled in techniques such as workload characterization, system measurement, interpretation of performance data, and system tuning, and are at least acquainted with basic statistical methods and with simulation.

Course Contents

- **Introduction to Queueing Theory:** Little's Law, Utilization Law, Kendall's Notations, Basic Relations for Birth-Death Processes, Open and Closed Queueing Networks, Product Form Networks, Mean-Value Analysis, Queueing Network Models of Computer Systems
- **Priority Queuing, Computer Communication Network, Multi-access Systems**
- **Overview of Queueing Network Modelling:** Modelling Study, Fundamental Laws, Queueing Network Models Input and Output
- **General Analytic Techniques:** Bounds on performance, Models with one or multiple job classes, Flow Equivalence and Hierarchical modeling
- **Representing Specific Subsystems:** Memory, Disk I/O, Processors
- **Parameterization:** Existing systems, Evolving systems, Proposed systems
- **Extended Applications:** Discrete Time Network, Fluid Approximations, Brownian Networks I, Brownian Networks II, Stability, Scheduling and Control

Teaching Methodology
Lectures would be interactive and it would cover the core concepts that are explained in the text and reference materials with adequate examples. Problem solving session will have conceptual and numerical questions that would aid in strengthening the queueing network modeling.

Text Book

1. Edward D. Lazowska, John Zahorjan, G. Scott Graham, and Kenneth C. Sevcik, "Quantitative System Performance, Computer System Analysis Using Queueing Network Models", Prentice Hall, Inc. Englewood Cliffs, New Jersey.
2. Raj Jain, "The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling", Wiley-Interscience, New York, April 1991.

Reference Books

1. F. Baccelli, P. Bremaud. Elements of Queueing Theory. Springer, 1991.
2. X. Chao, M. Miyazawa, M. Pinedo. Queueing Networks. Wiley, 1999.
3. H. Chen, D. D. Yao. Fundamental of Queueing Networks, Springer, 2001.
4. L. Kleinrock. Queueing Systems I & II, Wiley, 1975.
5. R. Serfozo. Introduction to Stochastic Networks, Springer 1999.
6. J. Walrand. Introduction to queueing networks. Prentice-Hall. 1989.

Course Description

Title of Course: Mathematical Modeling and Simulations Course Code: 14M14MA231
L-T Scheme: 3-0 Course Credits: 3

Objectives: In this course we have studied the Mathematical modeling which consists of applying your mathematical skills to obtain useful answers to real problems. Models are used in a very wide range of applications, some of which do not appear initially to be mathematical in nature. Learning to apply mathematical skills is very different from learning mathematics itself.

Learning Outcomes: Mathematical modeling and simulation are widely used in planning, security and communication, flow of water, electricity, gas and oil.. Students learn many mathematical modeling and simulation techniques, which are very helpful in analyzing and solving real life problems.

Course Contents:

Concepts of mathematical modeling; typical examples of Mathematical Modeling;
Mathematical Modeling using Difference Equations, Differential Equations
Random Numbers and Data
Introduction of Simulation Concepts in Discrete- event Simulation
Generation of Random-Number & Random-Variate with Applications
Queuing Models
Simulation Tools and Techniques, Input Modeling
Verification and Validation of Simulation Models,
Output Analysis for Simulation Model

Text Books

1. Jerry Banks, John S. Carson II, Barry L. Nelson and David M. Nicol, "Discrete Event System simulation".
2. Geoffrey Gordon, "System Simulation".
3. Dilwyn Edwards and Mike Hamson-Guide to Mathematical Modelling, Palgrace, 2007.

References

1. W. David, Randall P. Sadowski and Deborah A. Sadowski, "Simulation with ARENA".
2. Law. A.M. and W.D. Kelton, "Simulation Modeling and Analysis".

Course Description

Title of Course: Quantum Computation & Quantum Cryptography
Course Code: 14M14PH131
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students taking up this course are expected to be familiar with elementary calculus and matrix analysis. The necessary background in quantum mechanics and mathematical physics will be introduced as we go on in the course.

Objectives: This course is specifically designed to offer a pedagogical exposure for the students pursuing master's level studies in computer science and electronics. This newly emerging discipline provides many exciting opportunities for the practitioners of physics and engineering. In the first half of the course we intend to cover some fundamental concepts of quantum computation and quantum information theory. In the second half of the course, we will touch upon advanced topics, e.g., quantum cryptography, quantum error correction codes and quantum communication.

Course Contents:

Introduction & Overview: A brief historical review of basic ideas of classical computation and its scope and limitations. Basic definitions of quantum logic and quantum information. Basic ideas of classical information theory; measures of information (information content and entropy); Maxwell's demon, classical theory of computation; universal computer; Turing machine; computational complexity; uncomputable functions; shortcomings of classical information theory and necessity of quantum information theory. Stern-Gerlach experiment for illustration and existence of electron spin, basic idea of superposition of states.

Linear Algebra & Dirac Notation: Dirac notation and Hilbert spaces, dual vectors, linear operators. The spectral theorem, functions of operators. Tensor products, Schmidt decomposition theorem.

Qubits & The Framework of Quantum Mechanics: State of a quantum system, time-evolution of a closed quantum system, measurement in quantum mechanics. Pure and mixed states, density operator, partial trace, general quantum operators. Bloch-sphere representation of single qubit states, qubit rotations, single qubit gates.

Quantum Model of Computation: The quantum circuit model, single and multi-qubit operations, universal sets of quantum gates. Efficiency of approximating unitary transformations, implementing measurements with quantum gates.

Quantum Algorithms: Probabilistic versus quantum algorithms. Phase kick-back. The Deutsch and Deutsch-Jozsa algorithms. Quantum phase estimation and quantum Fourier transform, error analysis in arbitrary phase estimation. Finding orders, Shor's algorithm for order estimation. Quantum algorithms based on amplitude amplification, Grover's quantum search algorithm and related topics.

Quantum Entanglement & Teleportation: Mathematical and physical conceptions of quantum entanglement, entanglement distillation, entanglement of formation. Entanglement in pure and mixed states. No-cloning theorem for quantum states. Quantum teleportation and quantum communication.

Quantum Error-correction: Three qubit bit flip and phase flip codes. Theory of quantum errorcorrection, discretization of errors, independent error models. Construction of quantum codes, quantum circuits for encoding, decoding and corrections. Fault-tolerant quantum computing.

Preliminary Quantum Cryptography: Private key cryptography, quantum key distribution, privacy and coherent information, security of quantum key distribution.

References

1. Quantum Computation and Quantum Information, M.A. Nielsen and I.L. Chuang
2. An Introduction to Quantum Computing, P. Kaye, R. Laflamme and M. Mosca
3. Explorations in quantum computing, C.P. Williams and S.H. Clearwater
4. The Physics of Information Technology, N. Gershenfeld
5. Quantum Computing, M. Hirvensalo
6. The Principles of Quantum Mechanics, P.A.M. Dirac
7. Modern Quantum Mechanics, J.J. Sakurai
8. Mathematical Physics, S. Hassani, Springer-Verlag

2 year M. Tech. Course Curricula for Computer Science and Engineering							
M. Tech. IV semester (M4)							
S. No.	**Subject Code**	**Subject**	**Core/ Elective**	**L**	**T**	**P**	**Credits**
1		DE-V	Elective	3	0	0	3
2		DE-VI	Elective	3	0	0	3
3	14M19CI492	Seminar-II	Core	0	0	4	2
4	14M19CI491	Dissertation Part II**	Core	--	--	--	14
		Total					**22**

List of Electives for DE-V & DE-IV to be updated from time to time			**Core/ Elective**	**L**	**T**	**P**	**Credits**
DE-V							
1	14M14CI431	Computer Vision	Elective	3	0	0	3
2	14M14CI432	Machine Learning	Elective	3	0	0	3
3	14M14CI433	Grid Computing	Elective	3	0	0	3
4	14M14CI434	Swarm Intelligence & Applications	Elective	3	0	0	3
5	14M14CI435	High Performance Parallel Computing	Elective	3	0	0	3
6	14M14CI436	Software Architecture	Elective	3	0	0	3
DE-VI							
7	14M14CI437	Storage Networks	Elective	3	0	0	3
8	14M14CI438	Natural Language Processing	Elective	3	0	0	3
9	14M14CI439	Formal Language and Compilation	Elective	3	0	0	3
10	14M14CI440	Voice over IP	Elective	3	0	0	3
11	14M14CI441	Wireless Sensor Network	Elective	3	0	0	3
12	14M14CI442	Digital Forensics and Cyber Crime	Elective	3	0	0	3
13	12M1GMA133	Fuzzy Sets and Fuzzy Systems	Elective	3	0	0	3
14	12M1GMA134	Fuzzy Logic and Applications	Elective	3	0	0	3

Notes

1 Dissertation ** to be continued from III Semester
2 Final Evaluation of the dissertation will be based on the cumulative performance in all III and IV Semesters
3 It is desirable to have one publication from the dissertation

Course Description

Title of Course: Computer Vision **Course Code: 14M14CI431**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students must have good knowledge of "Computer Graphics and Image Processing and Applications".

Objectives:

- Strengthen higher level cognitive Skills of analysis, creation and evaluation.
- Strengthen Ability to representation of 2D and 3D Geometric Structures.
- Strengthen ability to express solution to problem clearly and precisely.

Learning Outcomes:
The student shall deepen the generic skills to representation of 2D and 3D Geometric Structures.

Course Contents:

Computer Vision Issues: Computer Vision Issues: Achieving simple vision goals, High-level and low-level capabilities, A range of representations,

Image Formation: Images, Image Model: Image Functions, Imaging Geometry, Reflectance, Spatial Properties, Imaging Devices: Photographic Imaging, Sensing Range, Reconstruction Imaging.

Early Processing: Recovering Intrinsic Structure, Filtering the image: Template, Finding Local Edges, Range Information from Geometry, Surface orientation from Reflectance Models, Optical Flow.

Boundary Detection: Hough Method for curve detection, Contour following, Motion - Motion Understanding, optical flow, understanding of image sequences.

Representation of 2D Geometric Structures: Two dimensional geometric structures, boundary representations, region representations, simple shape properties,

Representation of 3D structures: 3D: Surface representations, generalized cylinder representations, volumetric representations.

Pattern Matching: Graph Theoretic Algorithms, Multiple features, Classification.

Case Studies: Optical Character Recognition, Face Recognition.

Text Book

1. Robot Vision: B. K. P. Horn, MIT Press.

References

1. Machine Vision: David Vernon, online publication.
2. International Journal of Computer Vision, Springer.

Course Description

Title of Course: Machine Learning **Course Code: 14M14CI432**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students must have knowledge of "Artificial Intelligence".

Objectives: In this course we will study the basic component of an intelligence system i.e. machine learning, their functions, mechanisms, policies and techniques used in their implementation and examples.

Learning Outcomes: The students will have

1. Detailed knowledge of the concepts of machine learning.
2. Various application of machine learning in AI and different fields.

Course Contents

Introduction: Natural vs. machine learning, Types of Learning.

Inductive Classification: Concept Learning and General-to-Specific Ordering.

Decision Tree Learning: Properties, Top-down Induction, Entropy, Overfitting, Other Issues.

Artificial Neural Networks: Perceptron learning, Multilayer N/w, Backpropagation, Applications.

Experimental Evaluation: Inductive hypotheses, Types of Tests, Comparing learning Algorithms, Significance Testing

Bayesian learning: Naïve bayes, regression, Generative model and inference

Computational learning Theory: PAC model, Version Spaces, Complexity, Hypotheses Spaces, VC dimension

Instance based Learning: Distance Metrics, K-nearest neighbor and Variations.

Evolutionary learning: Genetic algorithms, Fuzzy models

Support vector machine: Separation, Classification, optimization, applications.

Other Learning: Reinforcement, statistical, unsupervised, temporal and explanation based learning

Analytical and Inductive learning: Analytical and Inductive learning

Text Books

1. Machine Learning: Tom Mitchell, McGraw Hill, 1997

References

1. Rich, Elaine Knight, Kevin, Artificial Intelligence, Tata McGraw Hill.

Course Description

Title of Course: Grid Computing **Course Code: 14M14CI433**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students must have already registered for the course, "Distributed System" and also must have extensive programming experience (e.g., in C/C++ and Java).

Objectives: The availability of powerful computers and high-speed networks as low-cost commodity components are changing the way computers are used. This leads to an increasing interest in the study of large-scale distributed systems: "cluster computing" in local-area-networks and "grid computing" in wide-area networks. The course will provide an in-depth introduction to Grid technologies and applications. The hands-on laboratory exercises will give participants practical experience with Grid middleware software and, based on it, the construction of Grid applications.

Learning Outcomes: By the end of this course, students will be able to-

1. Understand the need for and evolution of Grids in the context of processor- and data-intensive applications
2. Be familiar with the fundamental components of Grid environments, such as authentication, authorization, resource access, and resource discovery
3. Be able to design and implement Grid computing applications using Globus or similar toolkits
4. Be able to justify the applicability, or non-applicability, of Grid technologies for a specific application

Course Contents:

Overview and Motivation: The history and evolution of Grid computing, Situating Grid computing, the big picture, Basic concepts of Grid computing and Requirements, Large scale problem solving, Exemplars, Problems, trends and directions

Grid Architectures and Technologies: Nature of Grid architecture, Components of Grid, Layered Grid Architecture: Key Components: Grid Resource Allocation Management (GRAM), Grid FTP protocol, Grid Resource Information Service (GRIS). Resource infrastructure, Open Grid Services Architecture (OGSA), Globus Toolkit and Web Services.

Further Technologies: Storage Resource Broker, Legion, Condor, Grid Computing Environments, Resource allocation, sharing and discovery.

Peer-to-Peer Systems: Purpose, definition, characteristics, Types of P2P systems, Examples like Gnutella, Chord, Napster, KaZaA, Freenet, neuroGrid, JXTA, Relationship of P2P computing with GRID.

Security Issues in Grid environment: Introduction to GSI, Grid Applications, New Biology and the Grid, Data Intensive Grids, Applications for high-energy physics, Semantic Grid.

Text Books

1. The Grid: Blueprint for a New Computing Infrastructure (2nd edition) by Ian Foster (Editor), Carl Kesselman (Editor) Publisher: Morgan Kaufmann; 2nd edition (November 2003) ISBN: 1-558-60933-4.
2. Grid Computing: Making the Global Infrastructure a Reality by Francine Berman (Editor), Geoffrey Fox (Editor), Tony Hey (Editor) Publisher: John Wiley & Sons; (April 8, 2003) ISBN: 0-470-85319-0.

References

1. Grid Resource Management: State of the Art and Future Trends by Jarek Nabrzyski (Editor), Jennifer M. Schopf (Editor), Jon Weglarz (Editor) Publisher: Kluwer Academic Publishers; (September 2003) ISBN: 1-402-07575-8.
2. Ian Foster, Carl Kesselman: The Grid 2: Blueprint for a New Computing Infrastructure.
3. Maozhen Li, Mark Baker: The Grid: Core Technologies.
4. Borja Sotomayor, Lisa Childers: Globus Toolkit 4, Programming Java services.
5. Vladimir Silva: Grid Computing For Developers.
6. Bart Jacob, Luis Ferreira, Norbert Bieberstein, Candice Gilzean: Introduction to Grid Computing.
7. http://ws.apache.org/axis
8. http://www.oasis-open.org/specs/index.php#wsrfv1.2
9. http://www.gridforum.org
10. http://forge.gridforum.org/projects/ogsa-wg
11. http://www.globus.org/

Course Description

Title of Course: Swarm Intelligence and Applications **Course Code: 14M14CI434**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students must have good knowledge of "Artificial Intelligence".

Objectives: Swarm intelligence is a modern artificial intelligence discipline that is concerned with the design of multi agent systems with applications. Swarm intelligence is embedded in the biological study of self-organized behaviors in social animals, e.g., the collective behavior of social insects such as ants and bees, as well as flocks of birds and schools of fish. Instead of a sophisticated controller that governs the global behavior of the system, the swarm intelligence principle is based on many unsophisticated entities that cooperate in order to exhibit a desired behavior.

It will provide an overview of swarm intelligence as well as a collection of some of the most interesting latest applications from different domains. Furthermore, students will learn studying, processing and presenting scientific material about swarm intelligence through this seminar course.

Learning Outcomes: By the end of this course, students should be able to-

1. Understand Swarm Intelligence
2. Understand the application of Swarm Intelligence

Course Contents:

Introduction: Social Insects, Modeling Collective Behavior in Social Insects, Modeling as an interface, from algorithm to Robotics.

Ant Foraging Behavior and Application in Network: Overview, for Aging Strategies in ANT's, ANT Colony Optimization: The Traveling Salesman Problem, Other Applications of Ant Colony Algorithms to Optimization, Applications to Telecommunications Networks.

Cemetery Organization, Brood Sorting, Data Analysis, and Graph Partitioning: Overview, Cemetery Organisation and Larval Sorting, A model of Corpse Clustering and Brood Sorting, Exploratory Data Analysis, Graph Partitioning, Robotic Implementations.

Self-Organization and Templates: Application to Data Analysis and Graph Partitioning: Overview, the interplay of self organization and templates, Application.

Nest Building and Self-Assembling: Overview, Nest Building in social insects, Model of self-assembly, beyond biology.

Cooperative Transport by Insects and Robots: Overview, Cooperative prey retrieval in ANT's, Cooperative Transport by a swarm of robots.

References

1. Swarm Intelligence From Natural to Artificial Systems: Eric Bonabeau, Marco Dorigo, Guy Theraulaz, Oxford University Press 1999.
2. Swarm Intelligence, James Kennedy, and Russell C. Eberhart, Morgan Kauffman, 2001.
3. Ant Colony Optimization, Marco Dorigo, and Thomas Stutzle, MIT Press, 2004.

Course Description

Title of Course: High Performance Parallel Computing **Course Code: 14M14CI435**
L-T Scheme: 3-0 **Course Credits: 3**

Objectives: With the growing complexity of computer simulations and the availability of multi-core processors, parallel computing becomes important to all fields of science and technology. This course covers parallel high-performance computing on all levels: from the basics to high-level parallelism and grid computing. It is a hands-on course with practical programming exercises.

Learning Outcomes: Students will be able to-

1. Choosing the proper programming paradigm for an application
2. Shared memory implementation using OpenMP
3. Distributed memory implementation using MPI
4. Knowing parallel algorithms, data structures, and numerical solvers
5. Implementing loosely coupled applications on a grid of workstations

Course Contents:

Parallel programming paradigms, Vectorization, shared-memory and multi-core programming, OpenMP, Multi-threading, Message Passing Interface (MPI), Non-determinism in parallel programs, Parallel debugging, domain decomposition schemes, Communication scheduling methods, Parallel linear algebra and parallel solvers, data structures and abstractions, Parallel algorithms and libraries, Grid computing, Resource allocation models.

References

1. Parallel Programming, Second Edition, Wilkinson and Allen.
2. Parallel Computer Architectures: a Hardware/Software Approach. D.E. Culler, J.P. Singh, with A. Gupta. Ed Morgan Kaufmann. 1999.
3. High Performance Cluster Computing. R. Buyya. Ed. Prentice Hall. 1999
4. Cluster Computing White Paper. M. Baker, et al. 2001.
5. Using MPI, Portable Parallel Programming with the Message Passing Interface. W. Gropp, E. Lusk, A. Skjellum. Ed. MIT Press. 1999
6. Message Passing Interface Forum, http://www.mpi-forum.org/
7. The High Performance FORTRAN Handbook. Scientific and Engineering Computation Series C.H. Koelbel et al. The MIT Press, 1994.
8. UPC: Distributed Shared Memory Programming. T. El-Ghazawi et al. Wiley Series on Parallel and Distributed Computing, Wiley Interscience. 2005
9. Berkeley Unified Parallel C (UPC) Project. http://upc.lbl.gov
10. Parallel Programming in OpenMP. R. Chandra et al. Ed. Morgan Kaufmann, 2001.
11. The OpenMP API specification for parallel programming, http://openmp.org/
12. Overview of the MPI-IO Parallel I/O Interface. P. Corbett et al. Proceedings of the Third Workshop on I/O in Parallel and Distributed Systems, IPPS '95, Santa Barbara, CA. April 1995.

13. Improved Parallel I/O via a Two-phase Run-time Access Strategy. J. M. del Rosario et al. ACM Computer Architecture News. Volume 21(5), pages 31-38. December 1993.
14. ROMIO: A High-Performance, Portable MPI-IO Implementation, http://www.mcs.anl.gov/research/projects/romio/.
15. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. I. Foster, C. Kesselman, S. Tuecke, International J. Supercomputer Applications, 15(3), 2001.
16. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. I. Foster, C. Kesselman, J. Nick, S. Tuecke, Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002
17. Cloud Computing and Grid Computing 360-Degree Compared. I. Foster, Y. Zhao, I. Raicu, S. Lu, Grid Computing Environments Workshop, 2008. GCE '08 , vol., no., pp.1-10, 12-16 Nov. 2008
18. A history of cloud computing. A. Mohamed, March 2009.

Course Description

Title of Course: Software Architecture **Course Code: 14M14CI436**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisite: Students must have good knowledge of "Software Engineering".

Objectives: To give the understanding of software architecture and how it plays a central role for the success of a software system. The students will get knowledge of some well-known architecture patterns, and be able to design, construct and evaluate architectures for software systems. In addition, the students should get some understanding of how the developer's experiences and the technical and organizational environment will influence on the choice of architecture

Learning Outcomes:

- Know the definition of software architecture
- Understand the parameters of software architecture
- Learn how to develop quality characteristics that all stakeholders can believe in
- Know how to use the layering concept to build a software architecture
- Learn how to use patterns to jumpstart the software architecture
- Know how to use simple methods of validation and testing of the architecture
- Know when a formal method may be required in architecture design
- Understand how to build a documentation package for all stakeholders

Course Contents:

Introduction to Software Architecture: Definition of a Software Architecture, Purposes and Functions of a Software Architecture, The Business Case for a Software Architecture, The Overview of the Architecture Development Process, Problems in Building a Software Architecture,

Quality Attributes: Definition of a Quality Attribute, Characteristics of a Quality Attribute, Stakeholders and Quality Attributes, A Beginning set of Quality Attributes, Creating a Quality Attribute

Software Architecture Patterns: What is an Architecture Pattern?, 9 Patterns to Start, Patterns and the Application Domain, Patterns and Software Architecture, Reference Architectures.

Building the Architecture: The Software Hardware Split, The Layer Concept of Architecture Building, Build Me a Model!, Starting from a Pattern, Tactics for Specific Attributes,

Testing and Validating the Architecture: Review: The Functions of Testing and Validation, The Multiple Audience Problem, Simple Methods of Testing and Validation, A Peak at Formal Methods.

Documenting the Architecture: The Functions of Architecture Documentation, The Limits of Documentation, Using Views in Documentation, 7 Rules for Great Documentation

Case Studies and Final Project: This portion of the course incorporates the lessons learned throughout the program and builds in specific examples from your organization.

References

1. Bass, L., Clements, P. and Kazman, R., Software Architecture in Practice, 2003, Addison-Wesley.
2. Gorton, I., Essential Software Architecture, 2006, Springer.
3. Taylor, R., Medvidovic, N., Dashofy, E., Software Architecture: Foundations, Theory, and Practice, 2009, Wiley.
4. Eeles, P., Cripps, P., The Process of Software Architecgting, Addison-Wesley.
5. Clements, P., Bachmann, F., Bass, L., Garlan, D., Ivers, j., Little, R., Nord, R. and Stafford, J., Documenting Software Architectures: Views and Beyond, Addison-Wesley.

Course Description

Title of Course: Storage Networks **Course Code: 14M14CI437**
L-T Scheme: 3-0 **Course Credits: 3**

Objective: To strengthen the ability to design and implement the various aspects of storage networks which include the design model SAN, NAS, DAS, CAS, etc. and the various technologies like SCSI, Fibre Channel, INFINIBAND, IP Storage etc.

Learning Outcomes: Students will be able to-

1. About various components and protocols used for creating storage networks.
2. Know about various technologies existing for storage networks
3. Storage virtualization.
4. Existing techniques for management of storage networks

Course Contents

- **Introduction to Storage Technology**
  - Data proliferation,
  - Overview of storage infrastructure components
  - Evolution of storage
  - Information Lifecycle Management concept
  - Basic storage management skills and activities

- **Technologies for Storage Networks**
  - Disk Subsystems Overview
  - Architecture of Intelligent Disk Subsystems
  - JBOD: Just A Bunch of Disks
  - RAID & RAID Levels
  - Hot Sparing
  - Hard Disks and Internal I/O Channels
  - Caching: Acceleration of Hard Disk Access

- **I/O Techniques**
  - DAS, SAN, NAS, evolution,
  - Storage Area Networks (SAN): elements & connectivity
  - Fibre Channel SAN & Products
  - IP SAN Technology & Products
  - IP SAN elements, standards (iSCSI, iFCP, mFCP, FCIP and iSNS)
  - Migration from SCSI and Fibre Channel to IP storage
  - Network attached Storage: elements & connectivity

- **Management of Storage Network:**
  - Requirements of Management Systems

- Management Interfaces
- Standardized and Proprietary Mechanisms
- In-band & Out-band Management

- **Storage Virtualization**
  - The concept of storage virtualization
  - Storage virtualization on various levels of the storage network
  - Symmetric & Asymmetric Storage virtualization
  - Performance of SAN virtualization
  - Scaling storage with virtualization

Reference

1. "Storage Networks Explained", Ulf Troppens, Rainer Erkens, ISBN 0-470-86182-7, John Wiley & Sons.
2. "Storage Networks: The Complete Reference", R. Spalding, ISBN:0072224762, McGraw-Hill
3. "Storage Networking Fundamentals: An Introduction to Storage Devices, Subsystems, Applications, Management, and Filing Systems", Marc Farley, ISBN: 1-58705-162-1, Cisco Press.
4. "Designing Storage Area Networks: A Practical Reference for Implementing Fibre Channel and IP SANs, Second Edition", Tom Clark, ISBN: 0-321-13650-0, Addison Wesley.

Course Description

Title of Course: Natural Languages Processing **Course Code: 14M14CI438**
L-T Scheme: 3-0 **Course Credits: 3**

Objectives: This course provides an introduction to the theory of natural language processing (NLP). The creations of computer programs that can understand, generate, and learn natural language. We will use natural language understanding as a vehicle to introduce the three major subfields of NLP: syntax (which concerns itself with determining the structure of a sentence), semantics (which concerns itself with determining the explicit meaning of a single sentence), and pragmatics (which concerns itself with deriving the implicit meaning of a sentence when it is used in a specific discourse context). The course will introduce both knowledge-based and statistical approaches to NLP, illustrate the use of NLP techniques and tools in a variety of application areas, and provide insight into many open research problems.

Learning Outcomes: In this course, students will explore techniques for creating computer programs that analyze, generate, and understand natural human language. Upon completion of the course, students will have a broad understanding of what is involved in NLP generally and a good awareness of the issues related to syntactic, semantic, and pragmatics of language. Throughout the course, students will be exposed to state-of-the-art techniques for automatic language processing, including but not limited to automatic text understanding, machine translation, speech understanding, question answering, automatic summarization, intelligent information retrieval.

Course Contents:

Natural Language Processing: applications and key issues, The lexicon and morphology, Phrase structure grammars and English syntax, Part of speech tagging, Syntactic parsing, top-down and bottom-up parsing strategies, Semantics, Word Sense Disambiguation, Semantic parsing, Subjectivity and sentiment analysis, Information extraction, Automatic summarization, Information retrieval and Question answering, Machine translation.

Text Books

1. Speech and Language Processing, by D. Jurafsky and R. Martin, 2nd Edition.
2. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (second edition), D. Jurafsky and J. Martin.

References

1. Foundations of Statistical Natural Language Processing, C. Manning and H. Schütze.
2. Perl documentation

Course Description

Title of Course: Formal Languages and Compilation **Course Code: 14M14CI439**
L-T Scheme: 3-0 **Course Credits: 3**

Pre-requisite: Students must have good knowledge of "Discrete Mathematics".

Objectives:

- To understand the basic principles of finite tape and infinite automata theory, computability and complexity.
- To improve the ability to think abstractly, reason logically, write mathematical proofs and solve problems.
- To increase the ability to use mathematical language in a computer science setting.
- To develop problem solving ability using compiler programming

Learning Outcomes: Students will be able to -

- Design Automata, CFG, Turing Machine
- Test whether problem is computationally solvable or not.
- Work on JFLAP tool.
- Design and analyze a compiler

Course Contents:

Languages And Grammars: Vocabulary and language, Non languages Grammars, Derivations and language of a Grammar, Classification of Phrase Structure Grammars, Recognition of Context Sensitive Languages, Stages of Compilation.

Finite Automaton: Equivalence between Finite Automata and Regular Grammars, Extensions of finite automaton (Moore and Mealy Machines)

Regular Sets and Regular Grammars: Backward Deterministic Regular Grammars, Recognition Algorithm for Backward Deterministic Regular Grammars, Theorem: Conversion to backward deterministic form, Theorem: Closure properties of regular languages, Pumping Lemma for regular languages Equivalence between regular expressions and regular grammars, Theorem: Converting regular expressions to regular grammars, Theorem: Converting regular grammars to regular expressions, Another Formulation of Regular Languages, Construction of a Lexical analyzer

Context Free Languages(CFL): Derivations in Context Free Grammars Ambiguity, Recognition of context free languages, Chomsky normal form Theorem: Conversion to Chomsky normal form, Derivation In Chomsky Normal Form And Syntax Analysis, Importance of Chomsky Normal Form , Forward Deterministic Context Free Grammars, Pushdown Automata and context free Grammar

Pushdown Automaton: Top down parsing, Bottom-up parsing, Role of semantic analyzer, Writing a simple type checker.

Turing Machines: Turing Machine Model, Representation Of Turing Machine, Language Acceptability By Turing Machine, Design Of Turing Machine, Halting Problem Of Turing Machine, NP Completeness

Text Books

1. Introduction to Automata Theory, Languages and Computation: Hopcraft, J.E. and Ullman, J.D., 2nd edition, Pearson Education.
2. The Theory of Computer Science: Mishra, K.L.P. and Chadrasekaran, 2nd edition, Prentice Hall of India, 2001.

References

1. The Theory of Computation: Moret, B., Pearson Education, 2001.
2. Introduction to Languages and the Theory of Computation: Martin, John C. , 3rd edition, McGraw-Hill, Inc., New York, NY, 2003
3. An Introduction to Formal Languages and Automata: Linz, P, Narosa Publishing House, 2003
4. Introduction to the Theory of Computation: Sipser, M., PWS Publication.
5. Models of Computation: Savage, J. E., Addison-Wesley, 1998.
6. ACM Transactions on Computation Theory, ACM Journal on Theory of Computation, ACM SIGACT.
7. Compiler Design: Alfred V. Aho, Jeffrey D. Ullman, Pearson Education.
8. Compilers (Principles, Techniques and Tools): Alfred V. Aho, Jeffrey D. Ullman, Monica S. Lam, Ravi Sethi, Person Education.

Course Description

Course Title: Voice over IP **Course Code: 14M14CI440**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisites: Students must have Knowledge of "Computer Network & Introduction to Communication Systems".

Objectives: Voice over IP is designed for graduate students or practitioners to gain knowledge and develop a clear understanding of VoIP systems, components, standards, jargon and buzzwords, including packetized voice concepts, codecs and compression, RTP protocol, soft switches, gateways, servers, how SIP works and more on VoIP.

Learning Outcomes:
By the end of this course, students should be able to:

1. Understand the concept of VoIP and the protocols involved in VoIP
2. Understand the SIP.H.248 and different other Media gateway protocols

Course Content:

VoIP Systems, Components, Standards: Terminals, Voice in IP Packets, Soft switches/SIP Servers/Call Managers, Media Servers and Unified Messaging, Gateways, LANs and WANs, Key VoIP Standards.

VoIP Architectures: Internet Telephony, Managed-IP Telephone Service, VoIP for Businesses and Organizations, IP Phone Features and Uses.

VoIP Protocols: Session Initiation Protocols, MEGACO, H.248, Real Time Protocol, RTCP.

Understanding VoIP Technology and Networking: Overview, Voice Packetization, Measuring Voice Quality, Factors Affecting Voice Quality, Codecs, Compression and the G.729 codec, The VoIP Protocol Stack.

Carrier VoIP Networks, Services and Interconnect: Carrier Services Offered, MPLS and Quality of Service, VPNs and secure call paths, Session Border Controllers, Megaco/H.248 services, Types of managed services available, Gateway configurations and options, IP-level connectivity: cost savings

Voice over IP security: Internet Protocol (IP), User Datagram Protocol (UDP), Real-Time Transport Protocol (RTP), RTP Payload, Packet Analysis, Network Security Issues and Solutions

Text Book:

1. Daniel Minoli and Emma Minoli, Delivering Voice over IP Networks, 2nd Edition, John Wiley & Sons, Publication Date: August 23, 2002

2. Mark A. Miller, Voice over IP Technologies: Building the Converged Network, John Wiley & Sons. Publication, Mar 15, 2002

Reference Books-

1. Oliver C. Ibe, Converged Network Architectures: Delivering Voice and Data Over IP, ATM, and Frame Relay, John Wiley & Sons.
2. Bill Douskalis, Putting VoIP to Work: Softswitch Network Design and Testing, Prentice Hall.
3. Gilbert Held, Voice Over Data Networks: Covering IP and Frame Relay, McGraw-Hill Professional.

Reference Research Papers:

1. Khaled Salah, "On the deployment of VoIP in Ethernet networks: methodology and case study," Performance Evaluation of Wireless Networks and Communications, Vol.29, No. 8, pp.1039-1054, 2006
2. Garg S, Kappes M. Admission control for VoIP traffic in IEEE 802.11 networks. Proceedings of the IEEE Globecom'03, San Francisco, CA, U.S.A., pp.3514–3518, December 2003.
3. Qi Bi, Pi-Chun Chen, Yang Yang, Qinqing Zhang, "An Analysis of VoIP Service Using 1 EV-DO Revision A System," IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 24, NO. 1, pp.36-45, JANUARY 2006.
4. Arslan Munir," SIP-Based IMS Signaling Analysis for WiMax-3G Interworking Architectures," IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 9, NO. 5, pp.733-750, MAY 2010.
5. Jae-Woo So,"Performance Analysis of VoIP Services in the IEEE 802.16e OFDMA System With Inband Signaling," IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, VOL. 57, NO. 3, pp. 1876-1886, MAY 2008.

Course Description

Title of Course: Wireless Sensor Network **Course Code: 14M14CI441**
L-T Scheme: 3-0 **Course Credits: 3**

Prerequisites: Students must have knowledge of "Computer Network"

Objectives: This course will introduce students various aspects on ad-hoc/sensor networks, and expose them to the fundamental issues in designing and analyzing ad-hoc/sensor network systems. Students will study related technologies and standards ranging from networking, algorithms and other support. Of primary concern will be routing protocol, communication, data gathering, scheduling sleep cycles and computational challenges posed by these systems. Students will configure ad-hoc/sensor networks using QualNet, and study the performance of various protocols.

Learning Outcomes: This course will help students to identify the major issues associated with ad-hoc/sensor networks. Students will explore current ad-hoc/sensor technologies by researching key areas such as algorithms, protocols, hardware, and applications. At the end of this course students will gain hands-on experience through projects on ad-hoc/sensor network and be able to implement or develop algorithms involved in ad-hoc/sensor systems using QualNet simulator.

Course Contents:

Introduction of ad-hoc networks: Key definitions of ad-hoc networks, link quality, shadowing and fading effects, unique constraints and challenges, driving applications, wireless communications /radio characteristics.

Wireless Sensor Networks: Introduction, various industry standards, motes, scheduling sleep cycles, data gathering. Security.

LR-WPAN (IEEE 802.15.4): Introduction, PHY Specification, MAC Specification, ZigBee, Applications.

Routing Protocols: Issues in designing routing protocols, classification of routing protocols, Routing protocols.

Modeling sensor network: Introduction to QualNet simulator, creating scenario, configuring scenario, scenario statistics, packet tracer, file editor.

Applications of Ad-Hoc/Sensor Network and Future Directions.

Text Books

1. C. Siva Ram Murthy and B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols", 2nd Edition, Pearson Education, 2011.
2. Holger Karl and Andreas Willig, "Protocols and Architectures for Wireless Sensor Networks", Wiley India Pvt Ltd, 2011.

Reference Books

1. C.S. Raghavendra, Krishna M. Sivalingam and Taieb Znati, "Wireless Sensor Networks", Springer India, 2011.

2. Feng Zhao and Leonidas J. Guibas, “Wireless Sensor Networks: An Information Processing Approach”, Elsevier India Pvt. Ltd., 2011.
3. Nirupama Bulusu and Sanjay Jha, “Wireless Sensor Networks: A systems perspective”, Artech House, August 2005.
4. Anna Hac, “Wireless Sensor Network Designs”, John Wiley & Sons Ltd., 2003.
5. Mohammad Ilyas and Imad Mahgoub, “Handbook of Sensor Networks: Compact Wireless and Wired sensing systems”, CRC Press, 2005.
6. J. Yick, B. Mukherjee, D. Ghosal, “Wireless sensor network survey”, Computer Networks, Volume 52, Issue 12, 22 August 2008, Pages 2292-2330.
7. M. A. Yigitel, O. D. Incel, C. Ersoy, “QoS-aware MAC protocols for wireless sensor networks: A survey”, Computer Networks, Volume 55, Issue 8, 1 June 2011, Pages 1982-2004.
8. Dohler, M., Watteyne, T., Leung, K.K., "MAC Essentials for Wireless Sensor Networks", IEEE Communications Surverys & Tutorials, Vol. 12, No. 2, Second Quarter, 2010, Page(s): 222- 248.
9. Agoulmine, N., Maode Ma, Yanliang Jin, "Network lifetime optimization in wireless sensor networks" IEEE Journal on Selected Areas in Communications, VOL. 28, NO. 7, SEPTEMBER 2010, Page(s): 1127- 1137.
10. IEEE Computer Society, Part 15.4: Wireless Medium Access Control and Physical Layer Specifications for Low Rate Wireless Personal Area Networks (LR-WPAN), 2006.

Course Description

Title of Course: Digital Forensics and Cyber Crime **Course Code: 14M14CI442**
L-T: 3-0 **Course Credits: 3**

Prerequisite: Knowledge of Computer Networks, Operating Systems

Objectives: This course is intended to provide students with greater depth of study in a number of key topics in the area of computer security in society: cybercrime, computer and forensics, analysis. The course also involves significant practical work involving the development and use of forensics analysis tools.

Learning Outcomes:

- Understand digital forensics concepts, and explain their role in preventing various forms of fraud.
- Distinguish various types of computer crime, and use computer forensic techniques to identify the digital fingerprints associated with criminal activities.
- Know how to apply forensic analysis tools to recover important evidence for identifying computer crime.
- Understand legal perspectives of cyber crime

Course Contents:

Introduction to Digital Forensics, Cybercrime and Criminals, Digital Trails and Storage Systems, Hidden Evidences, Network Forensics, Image Forensics, Investigative Framework Methods, Cybercrime Methodologies, Digital Forensics Tools, Field Acquisition Methodology, Field Acquisitions of Electronic Media, Introduction to IT act.

Text Books:

1. Godbole, Nina and Belapure, Sunit, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley India, 2011
2. Nelson, Phillips and Enfinger, Steuart, Computer Forensics and Investigations, Cengage Learning, 2012

References:

1. Volonino, L and Anzaldua, R, Computer Forensics for Dummies, Wiley Publishing, 2008
2. Casey, E., Handbook of Computer Crime Investigation: Forensic tools and Technology, Academic Press, 2002
3. Johnson, T.A., Forensic Computer Crime Investigation, CRC press, 2006